EXHIBIT D

Certificate of Regist ation

This ertificate ıssued under the seal of the Copyright Office ın a rdance with ti e 17, *United S tes Code*, attests t at registra on has been made for the work identified below. he i for ation on this certificate has been made a part of the Copyright Office records.

Shira Perlmutter

United States Register of Copyrights and Director

Registration Number
TX 8-966-588
Effective Date of Registration:
April 28, 2021
Registration Decision Date:
May 27, 2021

Title

Title of Work: Wireline Payments Network

Completion/Publication

Year of Completion: 2019
Date of 1st Publication: February 01, 2019
Nation of 1st Publication: United States

Author

- **Author:** Vulcanize, Inc.
Author Created: text, Graphics
Work made for hire: Yes
Citizen of: United States

Copyright Claimant

Copyright Claimant: Vulcanize, Inc.
244 Fifth Avenue, #D281, New York, NY, 10001

Rights and Permissions

Organization Name: Butzel Long
Name: Jennifer Ann Dukarski
Email: dukarski@butzel.com
Telephone: (734)213-3427
Address: 301 East Liberty
Suite 500
Ann Arbor, MI 48104 United States

Certification

Name: Jenni er Dukarski
Date: April 28, 2021

Copyright Office notes: Regarding authorship information: Deposit contains copyrightable text, artwork only.

Wireline Payments Network

Rick Dudley and Eric Olszewski

February 2019

Wireline Payments Network

February 2019

Rick Dudley, Eric Olszewski

The mechanisms and concepts are derived from previous research and work done by Rick Dudley surrounding tendermint-based blockchains a cross-chain exchange protocols.

Abstract

In this document we introduce the Wireline Payments Network (the Network), which consists of protocols and runtime components to support low-latency, high-volume, micropayments across a network of microservices running on heterogeneous compute infrastructure (including existing cloud platforms and other decentralized systems).

Contents

Background and Motivation

History

Modern application development depends on a collection of centrally hosted servers, databases, and programs managed on public or private cloud infrastructure. Over the past decade, many businesses have moved their custom on-premises solutions to cloud platforms offered by companies such as Google, Amazon, and Microsoft. These platforms provide top-tier solutions to scalability, security, and many other infrastructural capabilities. Although the upside has been great, a number of businesses find themselves locked-in to these providers and unable to migrate their infrastructure; this leaves them subject to censorship, price discrimination, and surveilling of system and user data.

With the advent of decentralized systems, there is the possibility of secure and private peer-to-peer payments with digital currencies such as Bitcoin and peer-to-peer computation / storage with Ethereum. As these systems were recognized as solutions to the issues of censorship, contract lock-in, and surveillance, they quickly gained traction in the eyes of futurists and software enthusiasts. But, as these networks became more popular, the promise to deliver on a "world currency" or "world computer" fell into question as a result of scaling challenges inherent to the technologies.

To address this, developer communities focused on decentralized technologies have mobilized to create “Layer 2 Solutions” in the form of separate networks that facilitate transactions based upon on-chain escrows. In the case of Bitcoin, Lightning has become the de-facto scaling solution, whereas Ethereum has a number of projects such as Plasma and State Channels (e.g., Counterfactual) that are still being designed and implemented—and are vying for widespread adoption.

The first private implementations of systems like Plasma Cash in 2018 failed to address the problem of data unavailability, censorship by a single “operator”, and support for fungibility of assets. These systems also lack easy integration hooks for existing applications or tooling for participants within the network (e.g., identity/discovery of services). These shortcomings limit current Layer 2 Solutions to narrow use-cases which lack standardization for additional network tooling.

Motivation

The system incorporates a Service Registry that manages an extensible set of metadata—including identity, capabilities, contracts, and endpoints. The resulting "service mesh"[1] coordinates the discovery and connection of services and brokers the creation of point-to-point payment channels.

This network is a bonded proof-of-stake blockchain that implements a Lightning-compatible HTLC protocol to settle stable coin payments (DAI) on Ethereum. This enables point-to-point micropayments between dependent services, which enables low-latency, high-throughput transactions and supports all blockchains that support HTLCs.

[1] We use the term "service mesh" to mean a decentralized version of a related cloud pattern.

The resulting "usage" graph represents a network of relationships between users and interoperable services. This provides the basis of a reputational system, which mitigates Sybil attacks, and creates incentives for interoperability and composability for service developers and operators.

The initial version of this network will draw on an Ethereum-based stablecoin (DAI) for liquidity; additional sources of liquidity from disparate blockchains will be added in future iterations to provide for the first blockchain-agnostic micropayments solution. With this architecture, the need to integrate with specific cryptocurrencies by businesses will become anachronistic as participants will be able to accept the native stablecoin (WUSD) of the network and redeem it on any of the supported blockchains for their native equivalent.

Scope

Governance

A number of the mechanics serving as the Wireline Platform's foundation are based on governance processes that have yet to be decided by Wireline and WIRE holders. We provide a rough outline for some of these various processes in this document.

Medium of Exchange

The specifics on the exact medium of exchange or unit of account are out of the scope of this document. For the purpose of this document, we will be using a stablecoin derivative known as WUSD to serve these purposes in the Wireline Payments Network.

Ethereum-Based

While the Wireline Payments Network is able to use any HTLC-supporting blockchain as a parent chain or source of funds, this document will cover only the case of a singular parent blockchain (i.e, Ethereum).

Non-deterministic Computation

We assume that services provide non-deterministic computation (i.e., results cannot be independently provably verified by third-parties). Services call other services at their own risk, although their exposure is limited to the escrow required to initiate the payments channel. To mitigate obvious attacks, the registry tracks payment information, which may provide "trust anchors" as the basis of a reputation system.

Network Uptime

Although there are many techniques known for lowering the operational requirements of end-users and service providers, we assume that all entities connecting to the network will be doing so via nodes which receive and process transactions and blocks by participating in gossip (full nodes). We further assume these nodes will have 100% uptime.

Network Reputation

The Wireline Registry serves as both a discovery and reputation service for services registered to the network. This reputation system will be based upon service metrics (e.g., payments received) that serve as trust anchors that the network can attest to. The exact details of this reputation system as well as these trust anchors are out of this document's scope.

Securities Law

This is a technical specification to be reviewed by legal counsel - as such, we do not make any claims as to the legality of the network token or the incentivization mechanics supplied by the network to WIRE token holders.

Definitions

Wireline Payments Network

The Wireline Payments Network (WPN, or "network") is a federated Proof-of-Stake blockchain utilized as a parent chain-agnostic second layer micropayments network with a built in registry for payment-routing between participants within the network. This chain is implemented using Cosmos SDK and is run by a target number of (e.g., 6n +1 = 31) geographically dispersed validators.

Wireline Payment Channels

The Wireline Payment Channels (WPC) system is a Lightning Network inspired mechanism for near-instant, high-volume micropayments that removes the risk of delegating custody of funds to trusted third parties.

Wireline Registry

The Wireline Registry is a Distributed Hash Table (DHT) that manages metadata (including payments contracts) for services operating on the network.

Validator

Validators are nodes in the network responsible for managing consensus (in particular, block production) for state on the network. They are also responsible for operating the two-way pegs to other chains, network deposits, withdrawals, channel creation, and channel settlement.

Service Provider

Service providers are entities registered with the network that receive payments in WUSD from end users in exchange for providing infrastructural (e.g., computation and storage) or business services.

Parent Chain

The parent chain provides an external trust anchor for the funds used on the network. It must also provide the means for a collateralized stable coin. We assume the use of Ethereum (and DAI), but in principle any chain supporting HTLCs could be used.

Voucher

A voucher represents a claim against some asset on some parent chain. The funds backing the voucher exist in their entirety in an on-chain escrow. These escrowed assets can be released even if the network has failed. Given the initial focus on an Ethereum-based network, the only voucher detailed in this document is WUSD.

WUSD

WUSD is the native stablecoin of the network. WUSD are vouchers which are one-to-one redeemable for DAI on the Ethereum mainnet. WUSD is used as the primary unit of account in the network and to pay services within the network, ensuring that the cost of services remains predictable / stable.

Withdrawal Bond

This bond is used to combat the "nothing-at-stake" problem by network participants who have spent all their funds on the network and have nothing to lose for submitting a previously spent UTXO to the respective parent chain. In the case that an invalid withdrawal is attempted, this bond will be used to reimburse network participants who successfully dispute it. Addresses that have lost their withdrawal bond will be blacklisted.

WIRE

WIRE is the native token to the network which serves primarily for the facilitation of network governance, inter-service cohesion, and to thwart malicious network behavior. When a user stakes WIRE, they earn the right to vote.

Superblock

A superblock is a merkle tree of previous blockheaders in the network that is regularly submitted to each parent chain to ensure network data availability.

Tendermint

Tendermint is software for securely and consistently replicating an application on many machines. By securely, we mean that Tendermint works even if up to 1/3 of machines fail in arbitrary ways. By consistently, we mean that every non-faulty machine sees the same transaction log and computes the same state.

Cosmos SDK

The Cosmos SDK is a framework for building multi-asset Proof-of-Stake (PoS) blockchains, as well as Proof-Of-Authority (PoA) blockchains. The goal of the Cosmos SDK is to allow developers to easily create custom blockchains from scratch that can natively interoperate with other blockchains. We envision the SDK as the npm-like framework to build secure blockchain applications on top of Tendermint.

Sidecar

An infrastructure component run by services in the network, which is used to encode payment channel protocol metadata in HTTP request headers as well as monitor the status of open channels.

Overview

The Wireline Payments Network (WPN, or "network") is a Layer 2 federated (bonded Proof-of-Stake) blockchain using Tendermint consensus. It facilitates low-latency, high-throughput micropayments.

The network is operated by 6n + 1 (e.g., 31) validators. At the genesis of the network, each validator stakes a predetermined amount of WIRE on the payments network—and a predetermined amount of DAI on Ethereum. This stake can be slashed if a validator is found to be acting maliciously on either network.

To participate in the network, users send DAI to the network escrow contract on the Ethereum mainnet. After the transaction has finalized, the transaction is picked up by the network validators, and an Unspent Transaction Output (UTXO) is generated. The user's Wireline address is then issued an equivalent amount of WUSD (less some network fee) on the network.

> *Note: It is assumed that the network's cryptography is based on the secp256k1 curve and that a user's address on the network is the same as their address on Ethereum.*

Once WUSD has been issued, the user may open channels with users and services in the network using a Hashed-TimeLock Contract (HTLC). Network participants will then exchange micropayments over the Wireline Payment Channels (WPCs) and settle the channel by submitting the latest state to the HTLC on the network.

To redeem DAI on the Ethereum mainnet, users in the network must submit a withdrawal request to network validators. If the network has experienced a cataclysmic failure, users can submit each WUSD UTXO to the escrow on Ethereum themselves.

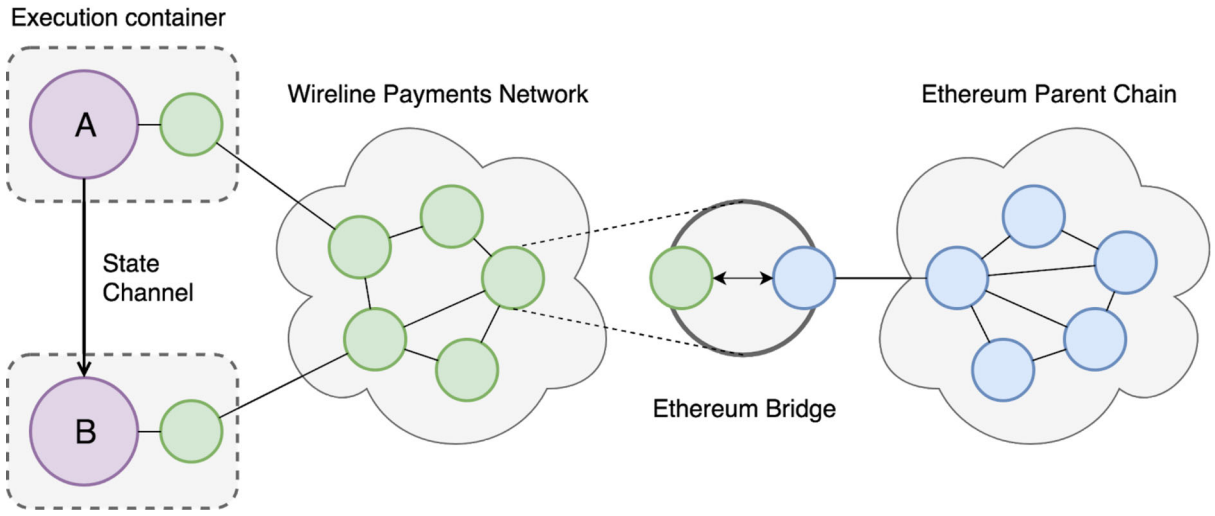

Diagram 1. Wireline Payments Network.

The above diagram illustrates the network, Ethereum Parent Chain, Ethereum Bridge, and service execution containers / state channels. The Ethereum Bridge serves to facilitate the exchange of ERC-20 DAI tokens for native WUSD tokens that are held on the network. Execution containers are used to facilitate micropayments and service orchestration via WPCs on the network.

Common Operations

Joining the Network

Validators

To join the network, there must be an open slot in the validator committee whereby a member of the prospective validator pool is pseudorandomly selected.

> *Note: All members of the prospective validator pool must be stake an equivalent amount of liquidity to current validators against all parent chains (includes staking of WIRE on the network) and be running a full node for each of the parent chains supported by the network. Nodes not running full nodes for each parent chain face higher risk of slashing due to latency in responses to parent chain activity.*

Service Providers

Members of the network wishing to act as service providers and receive WUSD in exchange for the services that they provide must undergo various Sybil-resistant measures (email confirmation, captchas, WIRE staking, etc.) with the Wireline Registry before it will route payments to them or make them discoverable to network participants.

In the case that service providers have dependencies on other fee-collecting network service providers, they will be required to escrow DAI on Ethereum to ensure fully-collateralized transactions.

Consumers

Consumers wanting to utilize network resources must escrow funds (in DAI) as well as a withdrawal bond (also in DAI) on the network escrow contract on Ethereum. After the escrow has been acknowledged on the network, a Consumer is free to interact with other entities on the network via WPCs.

WUSD Issuance

The network issues WUSD upon the escrowing of funds on the parent chain. After an entity escrows funds in a parent chain through some variant of the Lightning ERC20 Specification, the current network consensus round leader will then create a UTXO and wait a finalization period for the parent chain before committing the UTXO to the current block in the network. After the UTXO has been committed to the network, the new current consensus round leader will commit a proof of inclusion for the transaction to its respective parent chain. This inclusion proof will consist of the minted UTXO, the blockheader in which the UTXO was included, and the merkle path for the UTXO in the transaction merkle tree. This

inclusion proof will serve as the second phase of a two-phase commit process—finalizing voucher issuance on the network.

> *Note: All deposits, withdrawals, channel opens, and channel settlements made on the network are made less some fee taken by validators as well as a network tax.*

Voucher Transfer

Once a voucher has been issued on the network, it can be freely exchanged with other network participants via HTLCs serving as open channels between network participants. WPCs are used to interface with these HTLCs to facilitate such micropayments, with network participants each running their own WPC instances. Fungibility of vouchers is supported in this process via the existing UTXO model whereby derivative UTXOs are generated from the outputs of their parents and can be used for partial redemption on their respective parent chain.

> *Note: The network will not initially support bundled payments or subscription-based models because that would leave users open to potential lock-in with service providers providing degraded service. In the future, the network intends to support such functionality through the leveraging of TEEs and third-party auditors to ensure that service provider comply with SLAs required by various businesses and consumers.*

Voucher Redemption

Unless the network has failed, participants in the network are encouraged to redeem vouchers by leveraging the network validator set. Users who attempt to redeem vouchers directly when the network is not in a failure state will be able to redeem their funds but will be blacklisted from returning to the network and lose their withdrawal bond.

Upon receipt of a redemption / withdrawal request, validators will conduct a pruning / aggregation of their respective UTXOs before proceeding to withdraw to each parent chain. After the targeted UTXO for withdrawal has undergone some predetermined finalization period for the network, the leader of the current consensus round will submit a chain of UTXOs (or a reference to a chain of UTXOs) and a proof of inclusion for the target UTXO to the escrow contract on its respective parent chain.

> *Note: UTXOs will need to be tracked back to their originating parent chain and that UTXOs of disparate parent chains cannot be aggregated.*

After submission of a redemption, there will be a challenge period whereby members of the network may challenge invalid / stale redemption transactions. The challenge period is a set period of time in which any network participant can invalidate a redemption / withdrawal attempt by submitting a proof of inclusion for a child UTXO of the original target UTXO for withdrawal. A successful challenge will result in the submitting validator to have some percentage of their stake on the target parent chain slashed.

> *Note: Validators have a "weight" which is based upon the amount of their stake remaining. The process of slashing is not "all or nothing" in that a validator can be slashed multiple times before falling below a required validator stake threshold.*

If greater than ⅓ validators have fallen below a required validator stake threshold, the network will be in a faulted state and attempt to recover through the appointing of additional validators from the prospective validator pool at the end of the epoch. The faulted state will be evident to Ethereum as a “superblock” is required to be regularly submitted by the current consensus round leader to Ethereum and included in each Ethereum block. In the case where a contiguous threshold of superblocks have not been submitted to Ethereum, users may attempt to withdrawal their funds. If the withdrawal attempt succeeds, the network is then in failure mode and users will need to exit the network.

> *Note: Funds locked in WPCs during network failure may be redeemed through additional proof of inclusion process on Ethereum - there is also a default timeout for the escrows on Ethereum to ensure the return of funds to users.*

Network Tax

Network taxes in the network will be voted upon by validators within the network every epoch (one month) with the intention of meeting market demand by consumers whilst ensuring a fee structure conducive to the long-term maintenance and development of the network. Network taxes are a flat tax taken in WUSD on every deposit, withdrawal, channel open, and channel settlement processed by the network and are pooled in an on-chain escrow to be collectively distributed at each epoch.

Parent Chain Support

The process for supporting a parent chain begins with a proposal put forth by validators in the network followed by a voting period whereby ⅔ vote is required for the proposal to pass. If a proposal fails to get a ⅔ passing vote within a predetermined time-window, then it may be re-proposed re-evaluated by validators at a later epoch. In the case that a proposal passes, all validators will be required to run a full node for that chain as well as stake an equivalent amount of the decided-upon parent chain coin / token and demonstrate a valid liquidity proofs amongst themselves for their escrow.

In the case that a validator is unable to partake in this process (due to insufficient funds or other concern), they will be gracefully cycled out of the validator pool for members of the prospective validator pool. After all validators have provided liquidity proofs for their respective escrows on the new parent chain, they will begin accepting orders from said parent chain.

Slashing

Entities registering with the Wireline Registry will be required to deposit a predetermined amount of the network’s native token (WIRE) for the duration that they will be serving as a network service provider or validator. Deposits of validators will be slashed in the case that Byzantine behavior as exhibited through double signing or failure to sign, whereas withdrawal bonds of users will be slashed if they engage in the attempted theft of WUSD/DAI or in other to be determined malicious behavior.

> *Note: Conditions for slashing are out of the scope of this document and will be detailed in a case-by-case basis in further derivatives of this work.*

Faults

Validator Failure

In the case that a validator falls offline, the network will continue to process transactions at a degraded state. After coming back online, the previously faulted validator will process blocks but will be ineligible to propose new blocks until they have caught up with the tip of the blockchain. Validators who are offline will continuously be slashed until they either come back online or no longer hold stake in the network and need to be cycled out of the current validator set to be replaced by another validator in the following epoch.

> *Note: Failure to sign ⅓ of blocks in an epoch will cause a validator to be out of stake and cause the network to enter a degraded state.*

To account for regular validator maintenance, the network will allow for a predetermined set of validators to each separately be down for an extended time-period without risk of slashing. Validators who fail to come back online after this predetermined window will be slashed until they have been brought back online.

Network Failure

In the case of a faulted network—whereby □ ⅓ of all validators fall offline, consensus will halt. At this point, Wireline will attempt to intervene and inform users of the failure, but if <⅓ of the validators are not back online within a predetermined time-window, users will be able to redeem their funds on each parent chain - placing the network in failure state. At this point, the network is non-recoverable and users will be forced to withdraw their funds.

> *Note: Vouchers locked in WPCs at the time of network failure will be able to be redeemed in a similar manner to WUSD via a proof of inclusion for channel creation as well as submission of the final chanel state by both parties to the respective parent chain.*

Protocol

The Wireline Payments protocol implements a fully-collateralized, UTXO-based PoS chain.

It differentiates itself from most other scaling solutions in a number of ways.

- Many scaling solutions (all forms of Ethereum Plasma chains) offer a singular uncollateralized "operator" to serve as the consensus and commitment mechanism. By contrast, the network provides federation of fully collateralized nodes that decrease the likelihood of network failure and Byzantine behavior (i.e., validators have something at stake.)

- Network participants are required to escrow DAI in a withdrawal bond on Ethereum. Thus, fraudulent withdrawal attempts by participants can be punished on the parent chain (i.e., service operators and service consumers have something at stake).

- The network selectively applies fees to transaction types to provide resistance to "wash" attacks within the network which may lead network participants to infer higher quality of service due to greater transaction throughput for specific services within the network.

UTXO-Based Chain

The network uses UTXO transactions, rather than an account-based system, for purposes of scalability and future interoperability with UTXO-based second-layer scaling solutions. The use of UTXOs will also allow for easy interoperability with WPCs as our proposed transaction relaying mechanism.

> *Note: With accounts, users are exposed to replay attacks that can occur as a result of rebroadcasting earlier transactions. This is mitigated by maintaining sequence numbers on balances and requiring transactions to list the sequence numbers of the transactions which they take money from. Such a structure requires maintaining some data for every balance that was ever nonzero. For these reasons, the network will not be account-based.*

Vouchers / Escrows

Wireline allows for the trustless operation of validators through WUSD vouchers issued on the network that may be redeemed for DAI on the Ethereum Network. The process of depositing DAI into the network and redeeming WUSD vouchers for DAI on Ethereum are covered in the “Common Operations” section of this document.

> *Note: The only voucher which is presently supported by the network is WUSD (which may be redeemed for DAI on Ethereum), but this mechanism will be extended to other sources of liquidity based upon user-driven demand.*

Proof-of-Stake

Wireline uses a two-tiered proof-of-stake system to ensure proper network behavior among validators and network participants while encouraging delegation of resources toward continued network development and improvements by WIRE holders.

In this model, validators responsible for processing transactions on behalf of network participants must be staked on the network in WIRE as well as on the Ethereum Network in both ETH and DAI. Validators in the network will have their WIRE slashed in the event that they have double signed or are consistently unavailable. If evidence of this unavailability or byzantine behavior is present in the cryptographic artifacts which are regularly published to Ethereum, the faulty validator(s) will have their DAI on Ethereum slashed.

> *Note: A full detail of slashing conditions is provided in the Appendix of this document.*

The process of slashing is not “all or nothing” in that a validator can be slashed multiple times before falling below a required validator stake threshold. If a validator falls below this threshold, they will no longer be able to serve as a validator in the network and will be replaced at the end of the current epoch. As validators in the network are slashed, their weight (percentage of total validator stake) will decrease - lessening their influence on the consensus protocol which requires □ ⅔ of validator weight for block finalization.

Validators exhibiting proper network behavior will be rewarded through distribution of WUSD in the form of regularly collected network taxes as well as through having their staked WIRE grow with the rate of network inflation.

> *Note: This model resists attacks better than models that rely on the proper behavior of a singular "operator", often with nothing-at-stake, which allows for consistent malicious behavior without repercussion. A federation of validators also allows for greater network resilience with respect to attack vectors and network faults. This is most apparent in the case of node failure for existing scaling solutions using Proof-of-Authority; such failure would cause their chain to fail and force network participants to exit to Ethereum.*

Implementation

Payments Network Schematic

The Wireline Payments Network is built on top of the Wireline Registry, which uses the Cosmos SDK and Tendermint consensus.

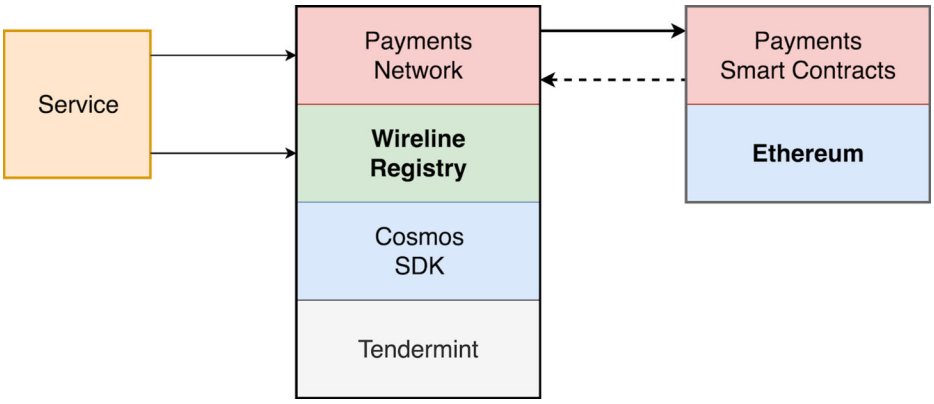

Diagram 2. Payments Network Schematic.

Payments Smart Contracts

The initial implementation of the network will only support Ethereum as a parent chain. On Ethereum, there will existing one or more smart contracts responsible for enabling the following functionality:

DAI

For the network to use DAI as a source of liquidity, the following operations for the corresponding use-cases will need to be supported:

- Escrow / depositing
    - Validator staking
    - UTXO generation
- Withdrawal / unlocking
    - Validator unbonding
    - Proofs of inclusion validation
    - Challenge / dispute periods

Slashing

The network's PoS implementation requires that smart contracts on Ethereum implement the ability to slash the following from poorly behaved network participants:

- User withdrawal bonds
- Validator stake

Checkpointing

In order to prove validators malicious behavior on the part of the validators, checkpointing will be leveraged with superblocks. These superblocks will contain headers from previous blocks in the network and be submitted to Ethereum at predetermined periods.

Wireline Registry

The Wireline Payments Network is based upon the Wireline Registry which serves as the foundational component of the network for service discovery. To become discoverable within the network, services must stake a non-trivial sum of WIRE - this stake will serve as a cost to services to provide some measure of sybil-resistance within the network. In subsequent iterations of the registry, stronger measures will be added for mitigating attack vectors stemming from Sybil attacks and the "nothing-at-stake" problem.

Wireline Payments System

The Wireline Payments System is composed of four main components: the Ethereum Parent Chain, the Ethereum Bridge, the Wireline Payments Network, and the Execution Containers for each service on the network.

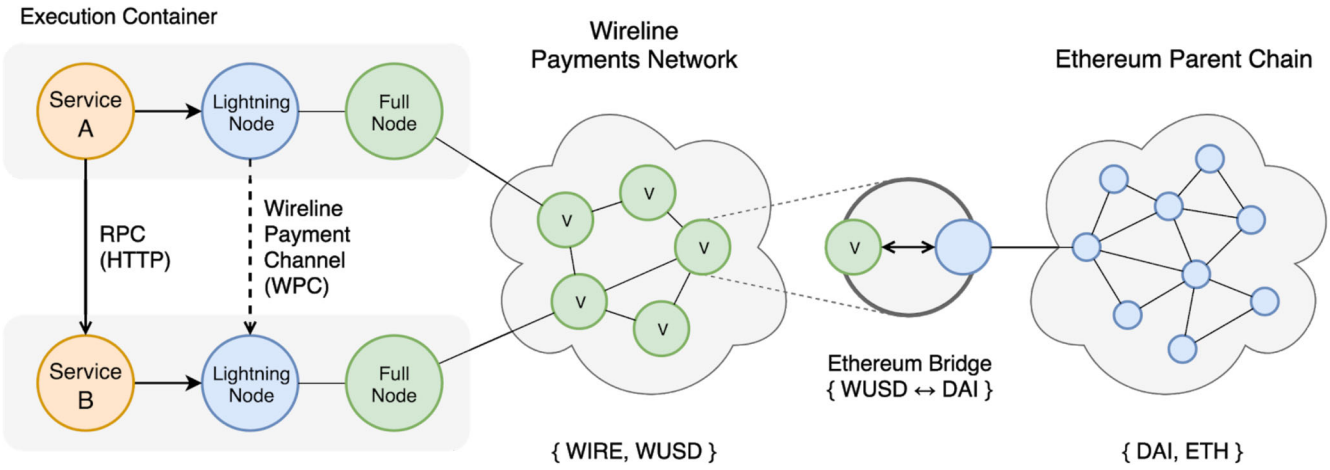

Diagram 3. Wireline Payments Network.

Ethereum Parent Chain

The Wireline Payments Network and the Ethereum Parent Chain exchange ERC-20 DAI tokens for native WUSD tokens that are held on the Wireline Payments Network via the Ethereum Bridge.

Ethereum Bridge

The Ethereum Bridge is one or more smart contracts that allow the escrow of DAI in return for an Ethereum-based UTXO (or, “ETHUTXO”). The implementation of this bridge will leverage past work on similar projects such as Drawbridge (Lightning / Ethereum bridge). Presently, this repository supports ETH and will need to be updated to support DAI as well as extended with the other needed functionality outlined in the previous section.

Wireline Payments Network

The Wireline Payments Network allows for the transfer of two token types between service consumers, service producers, and the network itself: the unit of account and medium of exchange for the WPCs (WUSD) and the inflationary native token for facilitating governance in the network (WIRE). The creation of WPCs will be facilitated in the same manner in which the Bitcoin Lightning Network creates its HTLCs for its payment channels.

Validators within the network will be responsible for the facilitation of all deposits into the network, transfers within the network, and withdrawals from the network. Failure to fulfill any of these responsibilities will result in slashing of validators and potential network faulting.

> *Note: Blocks in the network contain deposits (WUSD voucher creation), withdrawals (WUSD voucher destruction), channel creation, and channel settlement.*

Execution Containers

Services running on the Wireline Payments Network enter into payment channels via full nodes of the Wireline Payments Network. These full nodes are within the services' "trust boundary" (e.g., co-resident with the service's host machine). A detailed overview of the payment flow i provided in the next section.

Payment Flow

The following diagram illustrates the end-to-end operations involved in the facilitation of payments between services running on the network.

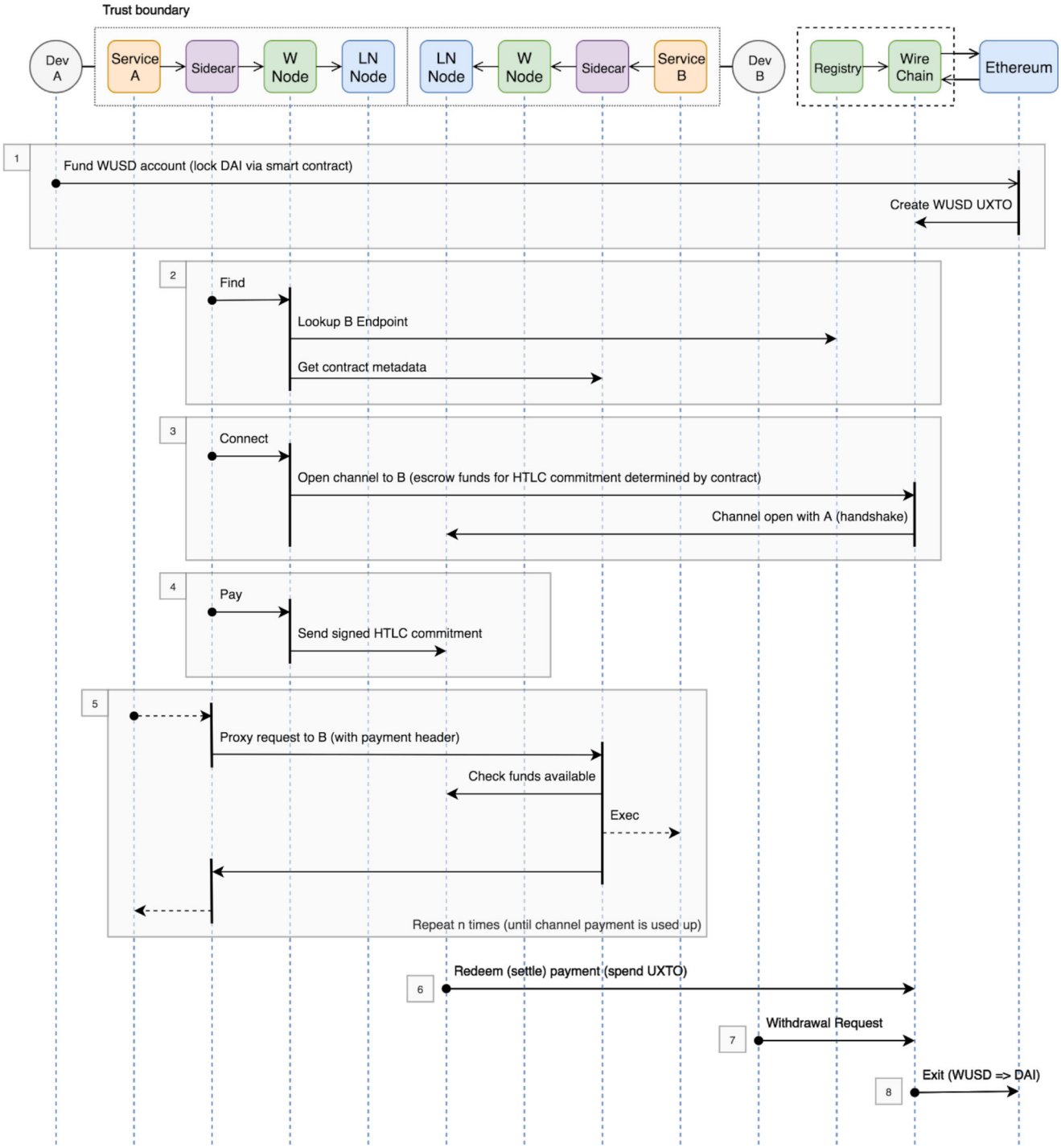

Diagram 4. Wireline Payments Sequence Diagram.

The diagram considers two services (A and B) operated by different developers. Each service executes within a trust domain that contains the following Wireline service nodes: a sidecar that proxies service and Wireline protocol requests, a Wireline Payments Network full node, and a Wireline Payments Channel interface. Each numbered sequence is annotated below.

1. Developer A deposits DAI via an Ethereum Lightning ERC20 smart contract. The Ethereum blockchain emits an event that is picked up by the network validator set after a set finalization period. After said finalization period, the current consensus leader issues the respective WUSD via a transaction sent to the WUSD contract on the Wireline Payments Network.

2. Service A attempts to find service B. It makes a GraphQL request to the Wireline Registry and retrieves the service endpoint. A's sidecar then retrieves metadata from B's "sidecar". This metadata includes the payment contract and identity requirements.

3. Service A submits a request to open a WPC via an on-chain escrow contract with Service B to the network validators. Service A escrows a portion of their existing WUSD to fund the channel.

4. Service A sends signed HTLC commitment to Service B so that B may add A to their internal channel registry.

5. Service A can now make requests to service B. This is coordinated via A and B's sidecar proxies, which encode payment channel protocol metadata in the HTTP request headers. B's sidecar checks that the channel is funded, then executes B's service and returns the results to A. Service B's sidecar maintains channel state which is comprised of signed requests from A. These requests continue until the contract limits are reached.

6. At any point, Service B can redeem its accumulated record payments with the Wireline Payments Network. This causes the deposited WUSD to be moved to B's account.

7. At any point, Service B can exchange WUSD for DAI by submitting a request to the network validators.

8. Network validators will submit the proof of inclusion for this withdrawal request after its parent superblock has been submitted to Ethereum. After the challenge period for the withdrawal has passed, the equivalent amount of DAI will be unlocked and redeemable by Service B on Ethereum.

Appendix A: Extensions

BLS Signatures

Finalization time and client-history size for transactions can be sharply decreased through the integration of BLS aggregate signature protocol—this would be implemented such that validators would sign every transaction and clients would only need to submit the signed transaction to the parent chain instead of a proof-of-inclusion with an interactive challenge period. This would involve significant engineering work on the PoS protocol to integrate such functionality (see example library implementations).

Appendix B -- References

Internal

- Payments PRD (placeholder)
- Payments Design

External

- LightningERC20 Implementation
- Example library implementations
- Proof of Inclusion Implementation
- Payment Channels
- Tendermint
- Cosmos SDK
- Slashing in Cosmos
- Drawbridge

Appendix C: Glossary

Wireline Payments System	The Wireline Payments System (WPS) is a protocol for facilitating highly-scalable micropayments between services, backed by a stablecoin on Ethereum.
Wireline Payments Network	The Wireline Payments Network (the Network, or WPN) is a federated Proof-of-Stake blockchain utilized as a second layer micropayments network on top off the Wireline Registry. This chain is implemented using Cosmos SDK using Tendermint consensus, and is run by a target number of (e.g., 6n +1 = 31) geographically dispersed validators.
Wireline Payment Channels	The Wireline Payment Channels (WPC) system is a Lightning Network-inspired mechanism for near-instant, high-volume micropayments that removes the risk of delegating custody of funds to trusted third parties.
Wireline Registry	The Wireline Registry is a Distributed Hash Table (DHT) that manages metadata (including payments contracts) for participating services.
Validator	Validators are nodes in the Wireline Payment Network responsible for managing consensus (in particular, block production) for state on the Network. They are also responsible for operating the two-way peg to Ethereum, and for deposits, withdrawals, channel creation, and channel settlement.
Non-Block-Producing Nodes	Non-Block-Producing Nodes are full Network nodes which neither participate in consensus nor have anything at stake. They are typically run by service providers within the author's trust boundary as a source of truth for current Network state.
Service Provider	Service providers are entities registered with the Network that receive payments in W:USD from their callers in exchange for providing infrastructural (e.g., computation and storage) or business services.
Parent Chain	The parent chain provides an external trust anchor for the funds used on the network. It must also provide the means for a collateralized stable coin. We assume the use of Ethereum (and DAI), but in principle any chain supporting HTLCs could be used.
Voucher	A voucher represents a claim against some asset on some parent chain. The funds backing the voucher exist in their entirety in an on-chain escrow. These escrowed assets can be released even if the network has failed. Given the initial focus on an

	Ethereum-based network, the only voucher detailed in this document is W:USD.
W:USD	W:USD is the native stablecoin of the network. W:USD are vouchers which are one-to-one redeemable for DAI on the Ethereum mainnet. W:USD is used as the primary unit of account in the network and to pay services within the network, ensuring that the cost of services remains predictable / stable.
Withdrawal Bond	This bond is used to combat the "nothing-at-stake" problem by network participants who have spent all their funds on the network and have nothing to lose for submitting a previously spent UTXO to the respective parent chain. In the case that an invalid withdrawal is attempted, this bond will be used to reimburse network participants who successfully dispute it. Addresses that have lost their withdrawal bond will be blacklisted.
WIRE	WIRE is the native token to the network which serves primarily for the facilitation of network governance, promoting inter-service cohesion, and thwarting malicious network behavior. When a user stakes WIRE, they earn the right to participate in governance.
Superblock	A superblock is a merkle tree of previous WPN blockheaders that is regularly submitted to Ethereum to ensure network data availability in the case of WPN failure.
Sidecar	An infrastructure component run by services in the network, which is used to encode payment channel protocol metadata in HTTP request headers as well as monitor the status of open channels.

Appendix B: Negative Incentives

We note there are two kinds of negative incentives, stake and bonds. While bonds are subject to loss in their entirety as a result of undesired behavior, stake maybe be taken ("slashed") incrementally over a longer period of time.

Withdrawal Bonds for WUSD-Holders

Who	Chain	Slashing Condition
WUSD	WPN	**WUSD-Holder attempts to withdraw WUSD -> DAI on Ethereum**

Holder		A WUSD-Holder attempting to withdraw WUSD -> DAI on Ethereum is making two claims: 1) The amount of WUSD being withdrawn has not already been spent elsewhere 2) The majority of validator weight will not post a superblock to ethereum within a given time period If either of the claims are invalid, the user loses their withdrawal bond and is banned from the system.

Staked Validators

Validators who lose enough stake during a certain epoch to put them below the minimum stake threshold lose their validation privileges at the end of the epoch.

Who	Chain	Slashing Condition
Validator	Ethereum, WPN	**WUSD-Holder Successfully Withdraws WUSD -> DAI on Ethereum** A WUSD-Holder attempting to withdraw WUSD -> DAI on Ethereum is making two claims: 1) The amount of WUSD being withdrawn has not already been spent elsewhere 2) The validators will not post a superblock to ethereum within a given time period If both of these claims are valid, the system enters an unrecoverable failure mode and all validators lose the entirety of their staked DAI. By extension, WIRE should be meaningless and valueless.
Validator	Ethereum, WPN	**Validator Double Signs (from Cosmos)** If someone reports on chain A that a validator signed two blocks at the same height on chain A and chain B, and if chain A and chain B share a common ancestor, then this validator will get slashed on chain A In this case the double signing validator loses a significant amount of stake, either down to the minimum threshold, or the entirety of their stake
Validator	WPN	**Validator Consistently Unavailable (from Cosmos)** If a validator's signature has not been included in the last X blocks, the validator will get slashed by a marginal amount proportional to X, and all the other validators will also get slashed by some small amount.

Notes

This section contains internal working notes and should be considered out of scope.

Write about relative block times, re-orgs, and transaction confirmations in addition to block confirmations.

> *Note: The description given here will be for an Ethereum-Tendermint two way peg system with an understanding that we may wish to support n-way pegs in the future.*

In order for the child chain to operate correctly it must be generating blocks at a higher rate than its respective parent chains, in this particular instance we will be aiming for 1 second blocktimes with finality on a child chain while ethereum will be providing 14 second block times with what we assume to be finality after 2048 block or approximately 8 hours (7 and 217/225ths hours).

The child chain validators require 2/3rds of the validator weight to sign a block before it is considered final. Every 14 (since the Wireline chain creates blocks every second and the Ethereum network does so every 14 seconds) child chain blocks, the leader of that round will create a “superblock”. A superblock is a merkle tree (or some other cryptographic primitive) that contains references to all the previous child chain blocks and the previous superblock. The parent chain genesis block of the child chain will count as a superblock.

We can optionally consider two types of blocks in our child chain: Blocks which contain data that does not need to be proven on the parent chain and block which contain data that needs to be proven on the parent chain.

The blocks in our child chain contain, deposits (which create vouchers), withdrawals (which destroy vouchers), and transfers (which send vouchers between entities on the child chain)

On the parent chain assets deposited by users wishing to use the child chain are pooled in an escrow controlled by the validators. If the user submits child chain vouchers (defined a set of UTXOs signed by the child chain validators), the user will be able to redeem those vouchers for the underlying escrowed asset from the parent chain escrow, regardless of the validators actions. It’s worth noting that is by definition would represent a failure of 2/3rds of validator weight and the child chain will be in a faulted state.

In normal operation, the user must submit a withdrawal request on the child chain, so that the two phase commit can be initiated first on the fastest chain and synchronicity between the two chains is maintained. The block which has been sealed (signed by 2/3rds of the validator weight) and contains the withdrawal request, represents a receipt that the validator set acknowledges the withdrawal. In an ideal implementation, the user should expect delivery of their asset on the parent chain after the next superblock submitted by the child chain validators to the parent chain smart contract is included in a block. Given that there is no real finality in Ethereum, we can also imagine a worse case withdrawal which occurs on the 2048th parent chain block (there might be an off-by-one error there). We can also imagine that processing constraints of the parent chain require that the user initiate the transfer of their

assets from the parent chain escrow to their parent chain wallet by submitting a proof of withdrawal inclusion to the parent chain.

The curent uncle rate in Ethereum is such that approximately 20% of the time a block at HEAD will not make it into the final chain. Practically this means that if we have 7 (6*n* + 1, where *n* is 0) validators on the child chain we should expect 1 (and 2/5ths[?]) of them to disagree.

Internal Review Tasks

1. Concerns that the payment network might have poor network effect (participants might move to another network, possibly even using the same code): doc should identify this attack, but its mitigation is out of scope, to be covered in another document. (Assigned to Ankur and Andrew).
2. Reputation signals: the document should cover the payments network's role in capturing signals that can from the basis of reputation metrics for participants (particularly service providers). Cover the scope of what kind of information can be captured and that it can be done under consensus (assigned to Andrew and Ankur).